WHAT IS CLAIMED IS:

1. An integrated circuit comprising:

a graphics pipeline; and

a frame buffer interface,

wherein the graphics pipeline comprises a shader coupled to a texture cache, the shader further coupled to the frame buffer interface,

wherein the shader stores and loads data to and from an external graphics memory using the frame buffer interface, and the shader is configured to store and load data to and from specific locations in the external graphics memory during a single pass through the graphics pipeline.

2. The integrated circuit of claim 1 wherein the shader executes instructions forming a shader program, the shader program executed during a plurality of passes.

3. The integrated circuit of claim 2 wherein one of the plurality of passes comprises:

executing a first plurality of instructions;

executing a read command, wherein data is read from a buffer in the external graphics memory and received by the shader using the frame buffer interface; and

executing a second plurality of instructions.

4. The integrated circuit of claim 2 wherein one of the plurality of passes comprises:

executing a first plurality of instructions;

executing a write command, wherein data is written to a buffer in the external graphics memory by the shader using the frame buffer interface; and

executing a second plurality of instructions.

5. The integrated circuit of claim 1 wherein the shader may write to and read from a first number of buffers located in the external graphics memory, the first number greater than two.

6. The integrated circuit of claim 3 wherein the first number of buffers are read-only buffers.

7. The integrated circuit of claim 5 wherein the first number of buffers are read/write buffers.

8. A method of generating a computer graphics image comprising:

executing a first plurality of instructions in a shader program, the shader program running in a shader in a graphics pipeline, the shader program executed during a plurality of passes through the shader;

executing a load command, wherein data is read from a first buffer by the shader; and

executing a second plurality of instructions in the shader program,

wherein the first plurality of instructions, the load command, and the second plurality of instructions are executed during a single pass through the shader.

9. The method of claim 8 wherein the shader may load data from a plurality of buffers, the plurality of buffers exceeding two buffers.

10. The method of claim 8 further comprising:

executing a store command, wherein data is stored by the shader in a second buffer; and

executing a third plurality of instructions in the shader program,

wherein the first plurality of instructions, the load command, the second plurality of instructions, the store command, and the third plurality of instructions are executed during a single pass through the shader.

11. The method of claim 10 wherein the shader may store data in and load data from a plurality of buffers, the plurality of buffers exceeding two buffers.

12. The method of claim 11 wherein the buffers are located in an external graphics memory.

13. The method of claim 12 wherein the shader stores data in and loads data from the buffers using a frame buffer interface.

14. An integrated circuit comprising:

a frame buffer interface; and

a graphics pipeline coupled to the frame buffer interface, the graphics pipeline comprising a shader coupled to a texture cache,

wherein the shader may access a plurality of buffer storage locations in a graphics memory using the frame buffer interface,

wherein the plurality of buffer storage locations exceeds two storage locations.

15. The integrated circuit of claim 14 wherein the graphics memory is not on the integrated circuit.

16. The integrated circuit of claim 15 wherein the shader may read data from the plurality of storage locations using the frame buffer interface.

17. The integrated circuit of claim 15 wherein the shader may read data from and write data to the plurality of storage locations using the frame buffer interface.

18. The integrated circuit of claim 17 wherein the shader reads and writes data to and from the plurality of storage locations using the frame buffer interface by using load and store commands in the shader program that include storage location identifications.

19. The integrated circuit of claim 17 wherein the shader reads and writes data to and from the plurality of storage locations using the frame buffer interface by using load and store commands in the shader program that include a direct reference to storage location addresses in the graphics memory.

20. The integrated circuit of claim 17 wherein the shader reads and writes data to and from the plurality of storage locations using the frame buffer interface by using load and store commands in the shader program that include indirect references to storage location addresses in the graphics memory.